CSE341 – Programming Languages 2023 FALL
Homework 4 Report

Ali ÇELİK 210104004219

Part 1

For part 1
"Given the current state, are there any delivery person available to pick and deliver a given object. If the object is already in delivery, return the person delivering it. Otherwise, print all the people that could make the delivery along with the total time to complete it."
is requested.

IDs are formatted as c# and p# for courier and packages.

I did not add screenshots of all facts. Below image shows rules that I defined.

```prolog
89   %Package's IDs are p#. Destination and route points are given. If package is not on route it have none in last part, else it have courier's ID.
90   package(p1, 15, admin, inst_x, low, none).
91   package(p2, 5, admin, library, med, none).
92   package(p3, 24, hall_A, inst_y, low, none).
93   package(p4, 32, cafeteria, soc_sci_bld, high, none).
94   package(p5, 10, library, eng_bld, med, c1).
95
96   %Rules
97   % Helper rule to find available delivery personnel for a package in a different location
98   available_courier(DestinationLoc, CourierID, State, ETA) :-
99       courier(CourierID, _, _, State, CurrentLocation),
100      route(CurrentLocation, DestinationLoc, ETA).
101
102  %To find who is carrying the package according to given PackageId
103  pack_courier(PackageId, CourierID):-
104      package(PackageId, _, _, _, _, CourierID).
105
106  who_can_deliver(PackageId):- %Function to see package's situation or see potential delivery guys.
107      pack_courier(PackageId, CurCourier),%Package's courier status is returned by pack_courier.
108      (CurCourier \= none ->
109          write("Package is already on route courier: "), write(CurCourier); %If package is on route already
110          %else
111          package(PackageId, PWeigh, PackageLocation, Destination, _, none), %Given package's details are gotten here.
112          available_courier(PackageLocation, CourierID, Status, ETA1), %If there is any available courier to come PackageLocation
113          (Status \= empty -> %If package is already on route
114              write(CourierID), write(" is already in delivery");
115              %else finde potential couriers
116              courier(CourierID, CourierCap, Whours, empty, _), %Given courier's details are gotten here.
117              route(PackageLocation, Destination, ETA2), %ETA is gotten here.
118              ETA is ETA1 + ETA2,
119              (PWeigh=<CourierCap -> %If weight is proper for courier
120                  (ETA=<Whours -> %If time is proper for courier
121                      write(CourierID), write(" is proper");
122                      %else
123                      write(CourierID), write(" is not proper because of working hours."));
124                  %else(If it is heavy)
125                  write(CourierID), write(" is not proper because of weigh capacity"))).
126
```

Part 2
In this part working on Iris datas is requested.
In pdf it is requested that returning Iris-<classname>. However, I returned them as only <classname>.

```prolog
3    %Knowledge base
4    iris(_, SW, PL, PW, Type):-
5        (PL=< 2.45 ->
6            (Type = setosa);
7            (PW =< 1.75 ->
8                (PL =< 4.95 ->
9                    (PW =< 1.65 ->
10                       (Type = versicolor);
11                       (Type = virginica)
12                   );
13                   (PW =< 1.55 ->
14                       (Type = virginica);
15                       (PL =< 5.45 ->
16                           (Type = versicolor);
17                           (Type = virginica)
18                       )
19                   )
20               );
21               (PL =< 4.85 ->
22                       (SW =< 3.10 ->
23                           (Type = virginica);
24                           (Type = versicolor)
25                       );
26                       (Type = virginica)
27               )
28           )
29       ).
30
31   %% Rule to identify given data belongs to which class
32   classify(SL, SW, PL, PW) :- iris(SL, SW, PL, PW, Type), write(Type), write("\n\n").
33   %%%%%%%%%%%%%%%      Decision Tree     %%%%%%%%%%%%%%%
```

Outputs

Part1

This is the output for 5 packages.

```
?- [delivery].
true.

?- who_can_deliver(p1).
c1 is already in delivery
true ;
c2 is not proper because of working hours.
true ;
c3 is proper
true.

?- who_can_deliver(p2).
c1 is already in delivery
true ;
c2 is proper
true ;
c3 is proper
true.

?- who_can_deliver(p3).
c1 is already in delivery
true ;
c2 is not proper because of working hours.
true ;
c3 is proper
true.

?- who_can_deliver(p4).
c1 is already in delivery
true ;
c2 is not proper because of weigh capacity
true ;
c3 is proper
true.

?- who_can_deliver(p5).
Package is already on route courier: c1
true.

?-
```

Part 2

Some outputs can be seen below.

```
?- [iris].
true.

?- classify(6.2,2.8,4.8,1.8).
virginica

true.

?- classify(6.4,2.8,5.6,2.1).
virginica

true.

?- classify(5.0,3.3,1.4,0.2).
setosa

true.

?- classify(5.7,2.8,4.5,1.3).
versicolor

true.

?- classify(4.9,2.4,3.3,1.0).
versicolor

true.

?-
```